3-D Electromagnetic Monte Carlo Particle-in-Cell Simulations on MIMD Parallel Computers

J. Wang and P.C. Liewer
Jet Propulsion Laboratory, California Institute of Technology
V.K. Decyk
University of California, Los Angeles

Computer particle simulation has become a standard research tool in space plasma physics research. A particle-in-cell (PIC) code simulates collisionless plasma phenomena by modeling a plasma as hundreds of thousands of test particles and following the evolution of the orbits of individual test particles in the self-collsistent electromagnetic field. Each time step in a PIC code consists of two major stages: the *particle push* to update the particle orbits and calculate the new charge and/or current density, and the *field solve* to update the electromagnetic fields. To study those problems that involve collisions between plasmas and neutral atoms, a particle-in-cell with Monte Carlo collisions (PIC-MCC) code can be used. In a PIC-MCC code, a Monte Carlo collision is incorporated into the particle *push* stage of a PIC code to calculate the collisonal effect on plasma particle orbits, While the PIC-MCC method allows one to study plasma phenomena from the very fundamental level, the scope of the physics that can be resolved in a simulation study critically depends on the computational power. The computational time/cost and computer memory size restricts the spatial scale, time scale, and number of particles that. can be used in a simulation. 'l'he cost of running 3-D electromagnetic PIC-MCC simulations on existing sequential super-computers limits the problems which can be addressed. Recent advances in massive] y parallel supercomputers have provided computational possibilities that were previously not conceivable.

A three-dimensional electromagnetic plasma PIC-MCC code has been developed for MIMD parallel computers. The code uses a standard relativistic particle push with Monte Carlo collision and a local finite-difference time-domain solution to the full Maxwell's equations. This code is implemented using the General Concurrent PIC (GCPIC) algorithm[1] which uses a domain decomposition to divide the computation among the processors[1]. Each processor is assigned a subdomain and all the particles and grid points in it. When a particle moves from one subdomain to another, it must be passed to the appropriate processors through interprocessor communications. To ensure that. the gatlicr/scatter steps and the collision step can be performed locally, each processor also stores *guard cells*, i.e. neighboring grid points surrounding a processor's subdomain which belong to another processor's subdomain. Guard cell information must be exchanged through appropriate processors for the current and electromagnetic field calculation.

This PIC-MCC code is used for very-large.scale simulations (i.e. over 10^8 particles) on three MIMD parallel computers: the 512-processor Intel Touchstone Delta, the 512-processor Intel Paragon, and the 128-processor Cray T3D. The performance of the code is analysed, and the results will be discussed.

1. P.C. Liewer and V.K. Decyk, J. Computational Physics, 85, 1989.